LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

1

LA-UR -87-227
(with follow-up)

LA-UR--87-227

DE88 009177

TITLE THE PERFORMANCE OF THE NEC SX-2 AND CRAY X-MP SUPERCOMPUTERS

AUTHOR(S) Olaf M. Lubeck
James W. Moore
Raul Mendez

SUBMITTED TO For distribution upon request

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO. 836 R4
ST. NO. 2629 5/81

The Performance of the NEC SX-2 and Cray X-MP Supercomputers

Olaf M. Lubeck

James W. Moore

Computing and Communications Division
Los Alamos National Laboratory

Raul Mendez

Department of Mathematics
Naval Postgraduate School

ABSTRACT

Since our first article on the Japanese supercomputers appeared in the December 1985 issue of Computer , we have conducted additional benchmarks of the new NEC SX-2 vector processor. We present these new results for the SX-2 and updated timings of the X-MP/48.

1. Introduction

This article is intended to be a follow-up to the original benchmark results in our first article on the Japanese supercomputers [1]. The same codes from the Los Alamos benchmark set have been executed on the NEC SX-2 and Cray X-MP during the period from December 1985 to October 1986. The original paper [1] contains a discussion of our measurement philosophy and a description of architectural differences among the Fujitsu, Hitachi, and Cray supercomputers. In this article, the emphasis is on the NEC machine and its performance relative to the Cray X-MP.

2. NEC SX-2 Architecture

Similar to the other Japanese machines, the SX-2 is a vector processor supercomputer that uses pipeline parallelism in both scalar and vector modes. It has some interesting differences, however. The SX-2 is really a two-processor system in which a control processor (CP) and an arithmetic processor (AP) are used asymetrically.

In general, the control processor executes the system and utility programs. Some of its functions are interactive terminal traffic, program development, and I/O.

The arithmetic processor executes the compute bound user programs. It consists of separate vector and scalar modules similar to both the Fujitsu and Hitachi machines (Fig. 1). The clock period of both modules is 6 ns compared with the X-

MP's 9.5 ns. (The new X-MP/416 has an 8.5 ns clock.) The vector processor has four separate sets of functional units. Because each set computes every fourth element of the same vector operation, the vector processor can perform 4 floating-point operations per cycle. The vector register capacity has 40 registers of 256 elements each. There are 8 load paths from and 4 store paths to memory. However, the total memory bandwidth is limited to 8 accesses per cycle and the load/store paths cannot operate simultaneously. Main memory is interleaved 256 ways.

The scalar processor has 128 registers and is interfaced to main memory through a 64K-byte cache. Table I compares the scalar floating-point operation times for the SX-2 and X-MP. The other Japanese supercomputers are also listed for reference. A detailed description of the NEC SX-2 is available in Ref. 2.

	Add	Multiply	Divide	Load
SX-2	6 CP 36 ns	9 CP 54 ns		49 CP 294 ns
X-MP	6 CP 57 ns	7 CP 66.5 ns	14 CP 133 ns	14 CP 133 ns
VP-200	3 CP 45 ns	4 CP 60 ns	28 CP 420 ns	
S810/20	2 CP 56 ns	4 CP 112 ns	21 CP 588 ns	

Table I. Scalar floating-point operation times. The divide operation in the X-MP is actually a reciprocal approximation.

3. Conducting the Benchmarks

The set of Los Alamos benchmarks that was executed on the Fujitsu and Hitachi machines was also run on the SX-2 and X-MP/48. Moreover, we added HYDRO, a major Lagrangian hydrodynamics code of the type in use at Los Alamos. Table II contains the percent vectorization and vector length for each of the benchmark codes. The benchmark set has been run on a broad range of both scalar and vector machines [3].

Code	Vectorization (percent)	Predominant vector length
BMK1	2	61
BMK4a	99	64,32,16,8,4,2
BMK5	0	-
BMK11	62	2056
BMK14	99	100
BMK21	0	-
BMK21a	18	35
BMK21b	0	-
BMK22	98	100
SIMPLE	93	62
HYDRO	98	100

Table II. Percentage vectorization as measured on a Cray-1 and predominant vector lengths for each of the benchmark codes.

As in the original paper, all tests were one-processor CPU tests. No I/O or throughput measurements were made. The NEC tests were accomplished in one week. Tuning changes were limited to minor Fortran revisions and the addition of compiler directives. The X-MP results were run using two compilers. The first is CFT1.14, the latest production version; the second is CFT77, a completely new compiler.

4. Scalar Performance

The relative scalar performance of the two machines can be ascertained from two non-vectorized Monte Carlo codes (BMK21 and BMK21b) and a scalar equation-of-state code (BMK5), as shown in Table III. The NEC machine is twice as fast as the X-MP using CFT1.14. However, the CFT77 compiler does a significantly better job with scalar optimization and the X-MP using CFT77 shows scalar performance comparable to the SX-2. It should be noted that the excellent scalar performance of the NEC machine has been reached despite the apparent disadvantage of large memory latency (49 clocks as opposed to the 14 clocks on the X-MP). We conclude that the NEC compiler has overcome this long memory latency by judicious use of cache and the 128 scalar registers and also by effective scheduling of instructions so that the latency is hidden by pipelining.

Code	SX-2	X-MP 1.14	X-MP 77
BMK21	1.6	3.2	2.0
BMK21b	67.7	132.6	79.4
BMK5	11.4	21.7	21.5

Table III. Times of selected scalar executions (sec).

5. Basic Vector Operations

Tables IV and V show the megaflop rates attained on the X-MP and SX-2 for a variety of vector operations and memory accesses. It has long been recognized that the Cray machines perform well on short vectors [4]. Fujitsu's VP-200 also has comparable short vector performance [1]. However, the data in Tables IV and V show that the NEC is typically better than the X-MP on short vectors (length=10) and is a factor of 2 to 4 faster on large vectors. This performance advantage over the entire range of vector lengths is significant.

Operation	10	50	100	200	1000
1. A(I)=B(I)+S	14.4	57.6	63.9	69.5	76.6
2. A(I)=B(I)+S (I=1,N,23)	9.6	39.3	52.0	61.4	77.7
3. A(I)=B(I)+S (I=1,N,8)	9.6	39.3	52.1	61.5	77.7
4. A(I)=B(I)*C(I)	14.0	53.5	58.6	61.6	68.7
5. A(I)=B(I)*C(I)+D(I)*E(I)	33.5	97.7	102.5	108.6	115.2
6. S=S+A(I)*B(I)	5.2	21.7	36.9	58.6	117.4
7. A(I)=B(J(I))+S	2.3	2.5	2.5	2.5	2.5
8. A(J(I))=B(I)*C(I)	3.1	3.5	3.5	3.5	3.5

Table IV. Rates (megaflops) on the X-MP for various vector operations as a function of vector length.

Operation	10	50	100	200	1000
1. A(I)=B(I)+S	21.9	109.7	219.3	340.1	369.5
2. A(I)=B(I)+S (I=1,N,23)	21.9	79.4	107.5	130.7	136.6
3. A(I)=B(I)+S (I=1,N,8)	16.9	62.7	96.6	132.3	237.7
4. A(I)=B(I)*C(I)	19.6	85.9	171.8	264.6	268.8
5. A(I)=B(I)*C(I)+D(I)*E(I)	38.8	171.2	342.5	492.6	530.8
6. S=S+A(I)*B(I)	17.6	74.7	123.0	178.2	549.9
7. A(I)=B(J(I))+S	10.3	33.2	43.4	49.8	51.6
8. A(J(I))=A(I)*B(I)	12.8	36.9	46.4	52.1	54.9

Table V. Rates (megaflops) on the SX-2 for various vector operations as a function of vector length.

6. Results from the benchmark codes

Table VI contains the timing data for our benchmark codes.

Code	SX-2	X-MP 1.14	XMP 77
BMK1	6.8	43.3	17.0
BMK4a	3.7	4.2	4.3
BMK5	11.4	21.6	21.5
BMK11	3.3	4.8	4.0
BMK14	.77	1.3	1.3
BMK21	1.6	3.2	2.0
BMK21a	3.9	8.3	4.0
BMK21b	67.7	132.6	79.4
BMK22	5.1	7.7	6.9
SIMPLE	2.4	5.8	7.0
HYDRO	10.1	17.7	-

Table VI Execution times (sec) for benchmark codes

Specific comments about each code follow:

1) BMK1 is an integer Monte Carlo code with virtually no floating-point instructions. The SX-2 is six times faster than the X-MP using CFT1.14 and 2.5 times faster than the X-MP using CFT77. Part of this difference is attributable to the fact that the SX-2 does 32-bit integer calculations. However, previous benchmarks also have pointed out the weakness of the Cray machines in integer calculations. For example, the CDC 7600 is faster than the Cray by a factor of 2 when executing this code.

2) BMK4a is an FFT code that is almost entirely vectorized. The times are comparable. The particular algorithm used for the FFT is not tuned to any machine and is not indicative of the true performance of any library routines that exist on the machines.

3) BMK5 is an excerpt from an equation-of-state code that is entirely scalar. The X-MP times are slower by a factor of 2.

4) BMK11 is a particle-in-cell code. The codes make use of the gather operation. The SX-2 executes about 50 percent faster than the X-MP.

5) BMK14 contains basic matrix operations on matrices of order 100. The NEC machine beats the Cray by a factor of almost 2.

6) BMK21, 21a, and 21b are Monte Carlo photon transport codes that are mostly scalar. The SX-2 is about twice as fast as the X-MP using CFT 1.14. The CFT77 compiler shows dramatic improvement in scalar optimization, and the times between the two machines are comparable.

7) BMK22 is a linear equation solver using Gaussian elimination. The SX-2 is roughly 70 percent faster than the X-MP. Dongarra [5] has independently measured a similar code in his LINPACK benchmarks. His measurements for equivalently tuned FORTRAN codes on matrices of order 100 are 43 mflops on the SX-2 and 24 mflops on the X-MP.

8) SIMPLE is a Lagrangian hydrodynamics code with heat conduction. The results in Table V are for a grid of 63 by 63. The SX-2 is again twice as fast.

9) HYDRO is another Lagrangian hydrodynamics application. It represents a significant fraction of the Los Alamos workload. It is much more realistic than SIMPLE and should be weighted more. On this code the SX-2 is about 70 percent faster.

7. Summary

The reader should note two important qualifications in the results of this benchmark:

1) All tests were one processor tests;

2) The results of this benchmark are highly dependent on the Los Alamos workload.

In all cases, the SX-2 executes our benchmarks as fast or faster than any other existing single-processor machine. It has the only processor that consistently outperforms the X-MP in all vector performance categories. It is 1.5 to 3 times faster on short vectors and 2 to 4 times faster on long vectors. The scalar performance is roughly comparable to the X-MP using the experimental Cray CFT77 compiler and twice as fast if the current Cray CFT1.14 production compiler is used.

The major advantage we see for the X-MP is that it and the Cray-2 are currently the only multiprocessor machines in the supercomputer class.

8. Acknowledgments

We wish to thank Tadashi Watanabe, Hiroshi Katayama, and the entire SX-2 staff at NEC for their support of our benchmark trip to the Fuchu NEC Plant.

9. References

[1] Olaf Lubeck, James Moore, and Raul Mendez, "A Benchmark Comparison of Three Supercomputers: Fujitsu VP-200, Hitachi S810/20 and Cray X-MP/2," IEEE Computer, Vol 18, No. 12, pp. 10-24, December 1985.

[2] Todanobu Furukatsu, Tadashi Watanabe and Ryozo Kondo, "NEC Supercomputer SX System," Nikkei Electronics, Vol 11, No. 19, pp. 237-272, 1984 (translated into English).

[3] M. L. Simmons and Harvey J. Wasserman, "Los Alamos National Laboratory Computer Benchmarking 1986," Los Alamos National Laboratory Technical report, November 1986.

[4] Bucher, I. Y. and James Moore, "Comparative Performance Evaluation of Two Supercomputers: CDC Cyber 205 and CRI CRAY-1," Los Alamos National Laboratory report LA-UR-81-1977, 1981.

[5] Jack J. Dongarra, "Performance of Various Computers Using Standard Linear Equations Software in a Fortran Environment, " Argonne National Laboratory report MSCD-TM-23, September 1986.

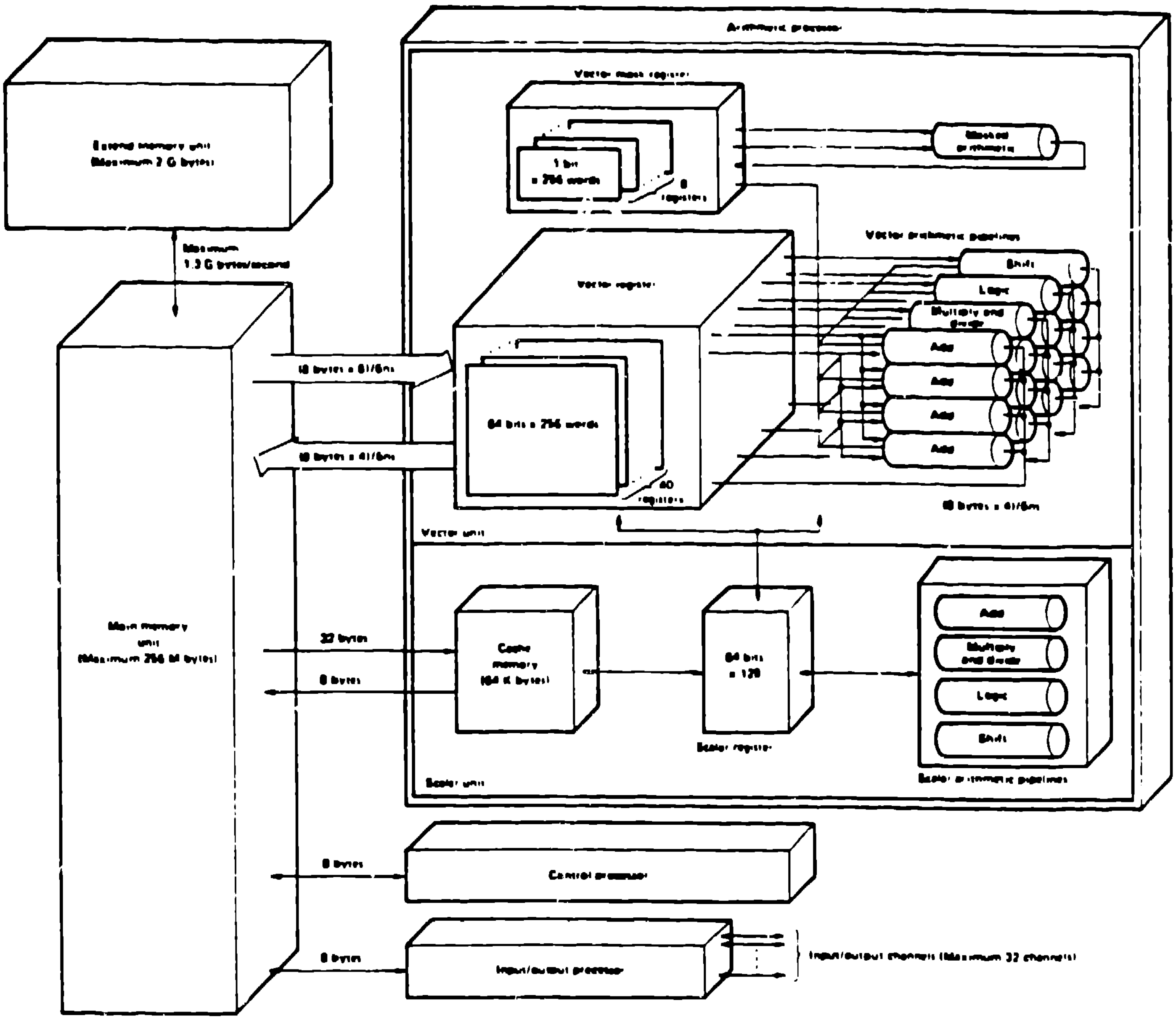

Figure 1. The SX-2 hardware configuration. The figure is taken from reference 2.

Los Alamos

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

memorandum

TO Ann Hayes, C-3, MS B265 DATE April 21, 1987

THRU James Moore, C-3 MAIL STOP/TELEPHONE B265/7-7028

FROM Olaf Lubeck/Margaret Simmons/Harvey Wasserman SYMBOL C-3-2127D

SUBJECT NEC SX-2 BENCHMARK

We did a follow-up benchmark of the NEC SX-2 at the Houston Area Research Center (HARC) in April 1987. The first benchmark was in December 1985. The purpose of the tests was to track any compiler improvements during that time and to perform some additional measurements of the machine. Included in the additional set was an attempt to measure basic I/O rates, execute a realistic out-of-memory code (WAVE) and execute an unclassified production code (MCNP).

Assistance was provided by Walt Colquitt of HARC and Mr. Sugimoto of NEC. Version 2.4 of the compiler was used. Table I contains a list of execution times comparing the December 1985 times with the April 1987 times. X-MP results are included for comparison. The results suggest that the compiler has been stable over the past year. Little change was observed in the execution times of most codes. We did note, however, that less tuning was required to get higher levels of vectorization. For example, HYDRO times in April 1987 are dusty deck (untuned) while the December 1985 times are tuned. The CFT77 and CFT1.14 compilers perform much worse on HYDRO than the NEC compiler. Additionally, the new compiler performs better on short vectors as evidenced by the significant change in BMK4a times. BMK1's increase in time using the latest compiler is anomalous. The increase stems from the fact that a loop was vectorized but runs faster in scalar mode.

There are a number of ways in which high-speed I/O may be carried out on the SX-2. The machine we measured was equiped with a one-GB Extended Memory Unit (XMU), what CRI calls an SSD. Both synchronous and asynchronous I/O are available but direct transfer between main memory and the XMU can only be carried out synchronously. Asynchronous data transfers to the XMU, as well as all transfers to/from disk, are performed by the Control Processor (CP). For this reason, I/O routines in large codes should be isolated, compiled for, and executed on the CP.

While we were not able to collect data for all these I/O modes, we did obtain good measurements of the high-speed (synchronous) XMU. Table II contains the raw measurements for this I/O mode. The times listed in Table II are cpu times. When doing synchronous I/O under dedicated conditions, these are equivalent to elapsed times. The SX-2 will allow measurement of elapsed time accurate only to the nearest second!

We model the I/O times as follows:

$T_{xmu}(l) = l/b\ [t_o + b*t_e]$

where:

l is the file size (64-bit words)
b is the record size (64-bit words)
t_o is the startup time
t_e is the element time

Figure 1 shows a fit of the model to the data (represented by X's). The fitted values of t_o and t_e are: t_o = 111 ns and t_e = 6 ns/word (the SX-2 has a cycle time of 6 ns). For comparison, Jordan and Bucher measured t_o = 270 ns and t_e = 9.5 ns/word on our CTSS operating system for SSD I/O. We should note that CTSS uses only one of two channels to the SSD. We should also obtain measurements using COS or UNICOS.

We are intrigued by the possibility of finding out more about I/O on the SX-2 as well as on other supercomputers. Consequently we would like to purchase time on the SX-2 at HARC to investigate this further as well as complete execution studies of MCNP and WAVE. We failed to get these two codes to execute due to the short time frame, although MCNP appears to compile correctly.

OL/MS/HW/jdm

Cy: C-3 File

Code	SX-2(12/85)	SX-2(4/87)	X-MP 1.14	X-MP 77
BMK1	6.8	10.8	43.3	17.0
BMK4a	3.7	2.8	4.2	4.3
BMK5	16.4	16.6	21.6	21.5
BMK11	3.3	3.3	4 8	4.0
BMK14	.77	.75	1.3	1.3
BMK21	1.6		2.0	
BMK21a	3.9	3.8	8.3	4.0
BMK21b	67.7		132.6	79.4
BMK22	5.1	4.9	7.7	6.9
SIMPLE	2.4	*	5.8	7.0
HYDRO	10.1	10.6	52.0	35.0

Table I Execution times (sec) for benchmark codes.

* compiler error

File Length(megawords)	Block Length(kilowords)	Time(secs)
1	50	.002427
1	10	.012105
1	2	.060447
1	.5	.243494
1	.25	.487806
1	.10	1 051628
1	.05	2 102167
10	500	.002432
10	100	.012120
10	20	.060520
10	5	242320
10	2.5	48699
10	[illegible]	1 212146
10	5	2 436952

Table II Raw High-Speed XMU I/O Rates

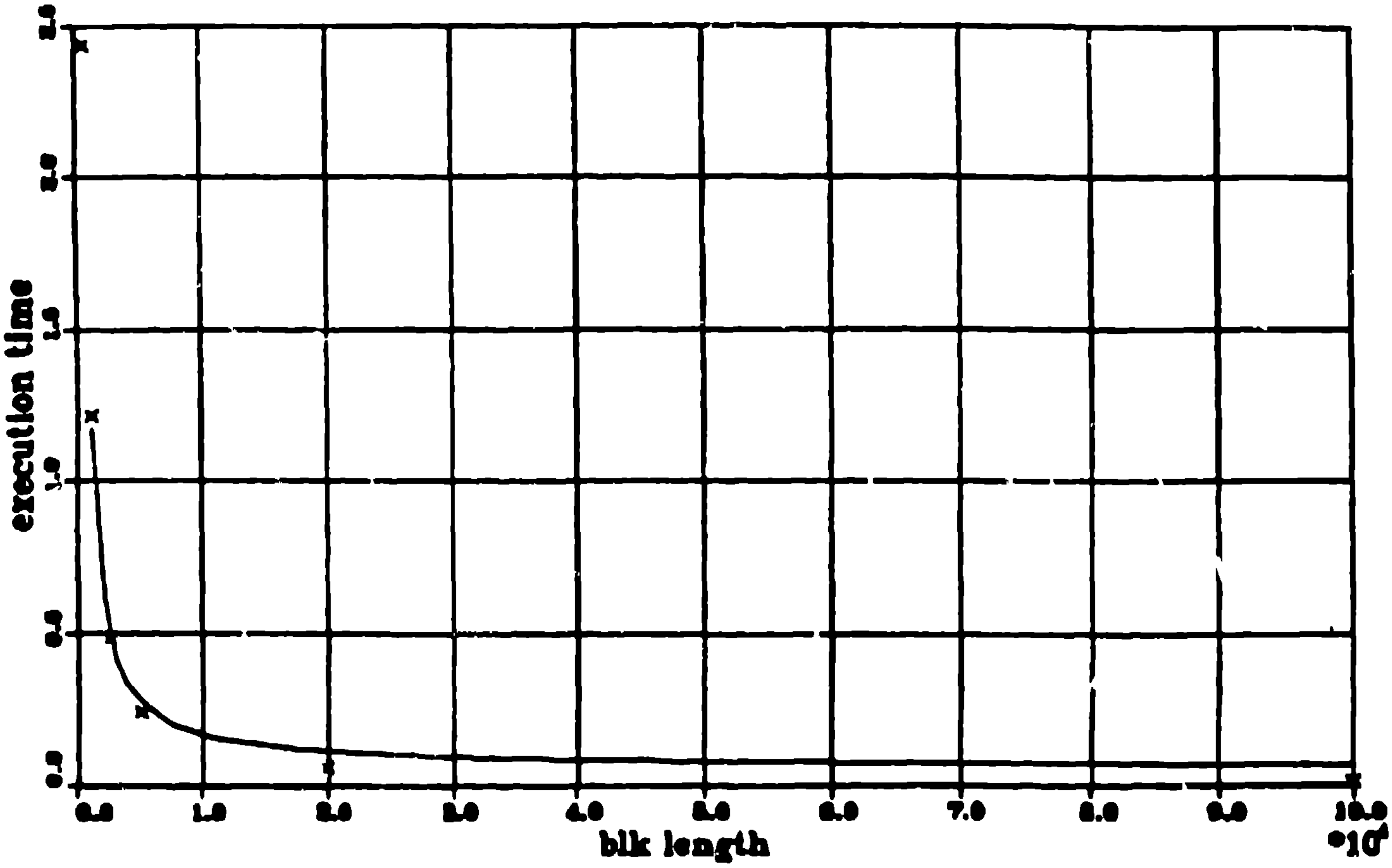

Figure 1.   Least Squares Fit of the High-Speed XMU I/O Times